Calhoun: The NPS Institutional Archive
DSpace Repository

Theses and Dissertations 1. Thesis and Dissertation Collection, all items

1965

Optimal filter design for sampled data systems with illustrative examples.

Jardine, Frederick D.

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/26669

Downloaded from NPS Archive: Calhoun

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

http://www.nps.edu/library

NPS ARCHIVE
1965
JARDINE, F.

OPTIMAL FILTER DESIGN FOR SAMPLED DATA SYSTEMS WITH ILLUSTRATIVE EXAMPLES

FREDERICK D. JARDINE

Library
U. S. Naval Postgraduate Sc
Monterey, California

OPTIMAL FILTER DESIGN
FOR SAMPLED DATA SYSTEMS WITH
ILLUSTRATIVE EXAMPLES

****

Frederick D. Jardine

OPTIMAL FILTER DESIGN FOR SAMPLED DATA SYSTEMS WITH ILLUSTRATIVE EXAMPLES

by

Frederick D. Jardine

Lieutenant Commander, Royal Canadian Navy

Submitted in partial fulfillment of
the requirements for the degree of

MASTER OF SCIENCE
IN
ENGINEERING ELECTRONICS

United States Naval Postgraduate School
Monterey, California

1965

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

PS ARCHIVE
368
[illegible], F.

OPTIMAL FILTER DESIGN
FOR SAMPLED DATA SYSTEMS WITH
ILLUSTRATIVE EXAMPLES

by

Frederick D. Jardine

This work is accepted as fulfilling
the thesis requirements for the degree of
MASTER OF SCIENCE
IN
ENGINEERING ELECTRONICS
from the
United States Naval Postgraduate School

ABSTRACT

This paper presents a method of optimal filter design for sampled data systems, based on the theory originally developed by R. E. Kalman. The first half of the paper deals with the theoretical development of mathematical models for linear, discrete dynamic processes and the optimal filter equations for such processes. The latter half discusses digital programming techniques for optimal filter design followed by two illustrative examples.

PREFACE

During the past several decades a fair amount of theoretical effort has been devoted to the study of problems which are of a statistical nature. Not the least important is the class of problems in communication and control which involves the separation of random signals from random noise, or the estimation of the states of a dynamic process based on noisy observations of a few of these states.

In several papers written since 1960, R. E. Kalman developed a theoretical approach for optimization of filters for the above mentioned class of problems. The theory is not all-embracing in that certain conditions must be satisfied before his technique can be employed.

The intention of this paper is to present a method for optimal filter design for sampled data systems, based on Kalman's approach. The first half of the paper deals with the theoretical development of mathematical model parameters for linear dynamic processes

and the optimal filter equations for such processes. The latter half discusses digital programming techniques for optimal filter design followed by two illustrative examples.

Library
U. S. Naval Postgraduate Sc
Monterey, California

TABLE OF CONTENTS

Section	Title	Page
1.0	Models for Random Processes	1
2.0	The Kalman Filter	12
3.0	Filter Design	20
4.0	Illustrative Examples	26
5.0	Bibliography	43

APPENDICES

I	Linear Differential Equations	44
II	The Stationary Filter	48
III	General Filters	54

CHAPTER I

MODELS FOR RANDOM PROCESSES

Before one can hope to achieve any amount of effective filtering, it is necessary that a fair amount of knowledge, about the physical phenomena to be observed, is known. For instance, if a sine wave buried in noise, is to be recovered, an apriori knowledge of the signal, i.e. frequency of the sine wave, is necessary. In addition if the statistics of the noise are known then optimum filtering can be achieved. It therefore becomes necessary to make a study of the message (signal) process before the construction of a filter is attempted. To maintain generality we will henceforth only consider random signals with the added stipulation that these signals are produced by linear dynamic systems excited by white noise.

1-1 LINEAR DYNAMIC SYSTEMS (CONTINUOUS TIME).

Since we are concerned only with linear dynamic systems a brief review of linear differential equations is in order.

A first order differential equation

$$\frac{dx}{dt} + \alpha x = u \tag{1.1}$$

has a solution (see Appendix I)

$$x(t) = e^{-\alpha t} x_0 + \int_0^t e^{-\alpha(t-\tau)} u(\tau)\, d\tau \tag{1.2}$$

where $e^{-\alpha t} x_0$ is the homogenous solution and $\int_0^t e^{-\alpha(t-\tau)} u(\tau)\, d\tau$ is the particular solution.

Consider now a set of first order differential equations, which define a linear dynamical system:

$$\frac{dx_i}{dt} = f(x_j, u) \tag{1.3}$$

or in vector notation

$$\underline{\dot{x}} = F(t)\, \underline{x} + D(t)\, \underline{u}(t) \tag{1.4}$$

where $\underline{x}$ and $\underline{u}$ are 1 x n column vectors and F and D are n x n matrices.

The solution (see Appendix I) to this set of equations is:

$$x(t) = e^{F(t)\, t} x_0 + \int_0^t e^{F(t)(t-\tau)} D(\tau)\, u(\tau)\, d\tau \tag{1.5}$$

or it may be written

$$\underline{x}(t) = \Phi(t,t_0)\,\underline{x}(t_0) + \int_{t_0}^{t} \Phi(t,\tau)\,D(\tau)\,\underline{u}(\tau)\,d\tau \qquad (1.6)$$

By definition we call the vector $\underline{x}$ the state of the system and $\underline{u}$ the input or control function.

Since all states (x_i) may not be observable we define the output of the system to be

$$\underline{y}(t) = H(t)\,\underline{x}(t) \qquad (1.7)$$

where $\underline{y}(t)$ is a p vector and H(t) is a pxn matrix. If all states were observable then H would be equal to the identity matrix I.

We can now represent the system in matrix block diagram form as shown in Fig. 1-1.

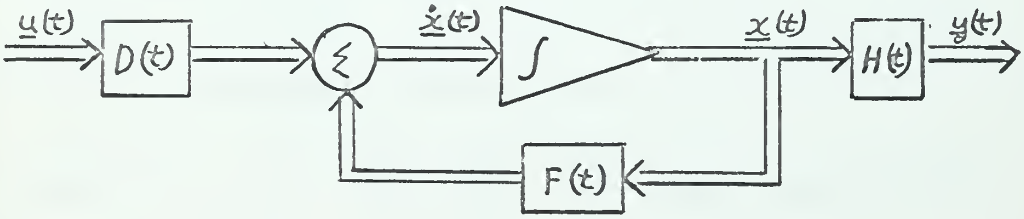

Fig. 1-1 Matrix block diagram of a linear dynamic system.

The integrator in Fig. 1-1 actually represents n integrators, one for each state of the system, while F(t) shows how the outputs of the integrators are fed

back to the inputs of the various integrators. Perhaps a look at a simple 2-state system at this time might clarify Fig. 1-1.

Given the linear dynamic system of Fig. 1-2, determine F(t), D(t), and H(t).

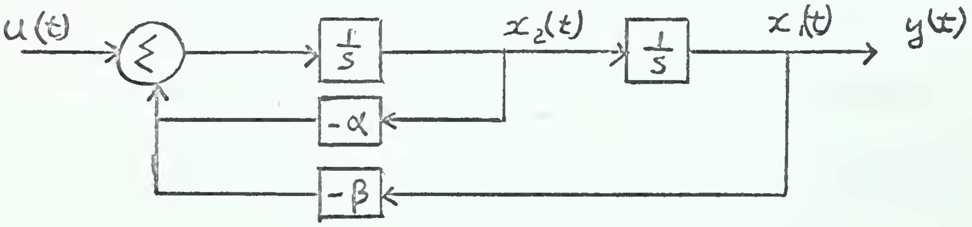

Fig. 1-2 A 2-state system

We can immediately write down the equations for the system as:

$$\dot{x}_1(t) = x_2(t) \tag{1.8}$$

$$\dot{x}_2(t) = -\beta x_1(t) - \alpha x_2(t) + u(t) \tag{1.9}$$

and our observable state(s)

$$y(t) = x_1(t) \tag{1.10}$$

It is immediately obvious that

$$F(t) = \begin{bmatrix} 0 & 1 \\ -\beta & -\alpha \end{bmatrix}, \quad D(t) = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \text{ and } H(t) = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

thus giving the vector differential equation

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\beta & -\alpha \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ u \end{bmatrix} \tag{1.11}$$

and
$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad (1.12)$$

1-2 LINEAR DYNAMIC SYSTEMS (DISCRETE-TIME).

If we specify the equations of a linear dynamic system in the form of difference equations then they are easily mechanized on present day digital computers. With this in mind the scope of this paper will be directed towards discrete-time situations.

In (1.6) we see that the continuous-time solution to a linear dynamic system is:

$$\underline{x}(t) = \Phi(t,t_0)\,\underline{x}(t_0) + \int_{t_0}^{t} \Phi(t,\tau)\,D(\tau)\,\underline{u}(\tau)\,d\tau \qquad (1.6)$$

If $\underline{u}(\tau)$ is held constant over the interval of integration then we obtain:

$$\underline{x}(t) = \Phi(t,t_0)\,\underline{x}(t_0) + \Delta(t,t_0)\,\underline{u}(t_0) \qquad (1.13)$$

where

$$\Delta(t,t_0) = \int_{t_0}^{t} \Phi(t,\tau)\,D(\tau)\,d\tau \qquad (1.14)$$

or more conveniently

$$\underline{x}(t+1) = \Phi(t+1,t)\,\underline{x}(t) + \Delta(t+1,t)\,\underline{u}(t) \qquad (1.15)$$

In (1.15) we assume a sampling period of one time unit. A block diagram of the linear discrete-dynamic system is shown in Fig. 1-3.

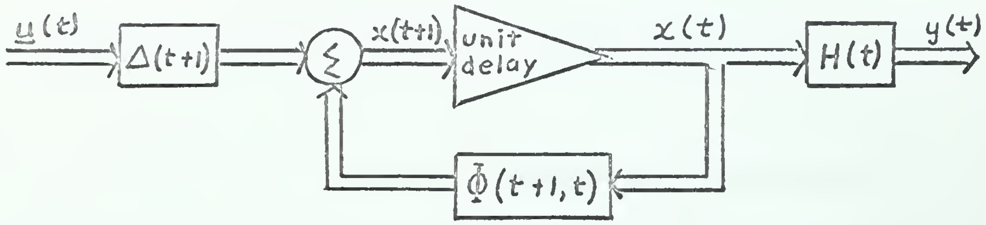

Fig. 1-3 Matrix block diagram of a linear discrete-dynamic system.

1-3 DETERMINATION OF MODEL PARAMETERS.

The matrix $\Phi(t+1,t)$ occuring in (1.6) and (1.13) is called the TRANSITION matrix and has the following properties:

$$\Phi(t_0, t_0) = I \quad \text{(Identity Matrix)} \tag{1.16}$$

$$\Phi(t_2, t_1)\,\Phi(t_1, t_0) = \Phi(t_2, t_0) \tag{1.17}$$

$$\frac{d\,\Phi(t, t_0)}{dt} = F(t)\,\Phi(t, t_0) \tag{1.18}$$

(1.16) and (1.17) are fairly obvious and (1.18) is obtained by setting $\underline{u}(t)$ to zero and differentiating (1.6). These properties can be useful in checking the accuracy of analytic expressions for the Φ matrix.

If the F matrix is constant then the transition

matrix elements depend only on the time difference $t-t_o$ and can be calculated from the following expression:

$$\Phi(t,t_o) = e^{F(t-t_o)} = \sum_{i=0}^{\infty} \left[F(t-t_o)\right]^i / i! \qquad (1.19)$$

A second (and easier) method for obtaining $\Phi(t,t_o)$ is by the use of signal flow techniques and the application of unit impulses to the input of selected integrators. This method will be demonstrated elsewhere in this paper.

The Δ matrix may be obtained by performing the integration in (1.14) or by the second method mentioned above for the Φ matrix.

1-4 THE GAUSS-MARKOV PROPERTY.

A large number of physical phenomena possess the Markov property. Basically it means that the best estimate of a future state of a process can be made without the knowledge of all the past history of the process. In a strict sense it implies that the best estimate of a future state can be derived from the last observation of the states. A very trivial example

would be the motion of a particle with a constant velocity vector. Given the best estimate of the present position and velocity of the particle one can formulate a best estimate of position and velocity for any time in the future. In fact the output from any linear dynamic system is Markovian. If $\underline{u}(t)$ is set equal to zero in (1.15) then this property may be expressed mathematically as:

$$\underline{x}(t+1) = \underline{\Phi}(t+1,t)\,\underline{x}(t) \qquad (1.20)$$

If $\underline{u}(t)$ is a gaussian random vector then the sequence of random vectors ...$\underline{x}(t-1)$, $\underline{x}(t+1)$, ... generated by (1.15) is known as a gauss-Markov sequence. The stipulation that $\underline{u}(t)$ is gaussian implies that the sequence ..., $\underline{u}(t-1)$, $\underline{u}(t)$, $\underline{u}(t+1)$, ... are normally distributed random vectors such that the cross-variance matrix:

$$\mathrm{COV}\left[\underline{u}(t_1),\ u(t_2)\right] = 0 \qquad \text{for } t_1 \neq t_2 \qquad (1.21)$$

i.e. $\underline{u}(t_1)$ and $\underline{u}(t_2)$ are independent. In addition the random vectors are completely defined by specifying their first and second order moments. i.e. $E(\underline{u}(t))$ and $E(\underline{u}(t).\underline{u}(t))$. For the purposes of this paper $\underline{u}(t)$

will be assumed to have zero mean;

i.e. $E[\underline{u}(t)] = 0$ for all t (1.22)

$E(\underline{u}(t) . \underline{u}(t)\ )$ is called the auto-covariance matrix of the vector $\underline{u}(t)$ and will be denoted by $U(t)$, i.e:

$$E[\underline{u}(t) \cdot \underline{u}(t)^T] = U(t) \tag{1.23}$$

or

$$\mathrm{cov}[\underline{u}(t)] = U(t)$$

Considering now the state of a process, we assume that the initial state, $\underline{x}(t_o)$, is a gaussian random variable of zero mean and arbitrary variance. By repeated application of (1.15), we see that future states, ..., $\underline{x}(t-1)$, $\underline{x}(t)$, $\underline{x}(t+1)$, ... will also be gaussian random variables, since they are obtained by linear combinations of gaussian random variables.

In probability terminology we may now define the Gauss-Markov property. Since $\underline{u}(t_1)$ and $\underline{u}(t_2)$ are independent for $t_1 \neq t_2$, then the conditional probability distribution of $\underline{x}(t)$ is dependent only on the previous state, i.e:

$$P(\underline{x}(t) \leq \underline{\eta} \mid \underline{x}(t-1), \underline{x}(t-2), \underline{x}(t-3), \cdots) \tag{1.24}$$
$$= P(\underline{x}(t) \leq \underline{\eta} \mid \underline{x}(t-1))$$

Where $\underline{\eta}$ is an arbitrary vector.

1-5 THE COMPLETE MODEL

In Fig. 1-3 the observable output vector $\underline{y}(t)$ cannot be measured with infinite accuracy and therefore to complete the model for random processes (with previously mentioned restrictions), a source of measurement noise must be added. This is illustrated in Fig. 1-4 where $\underline{v}(t)$ is a noise vector having the same dimensionality as $\underline{y}(t)$. $\underline{v}(t)$ is white noise (gaussian), which we assume to have zero mean with arbitrary variance:

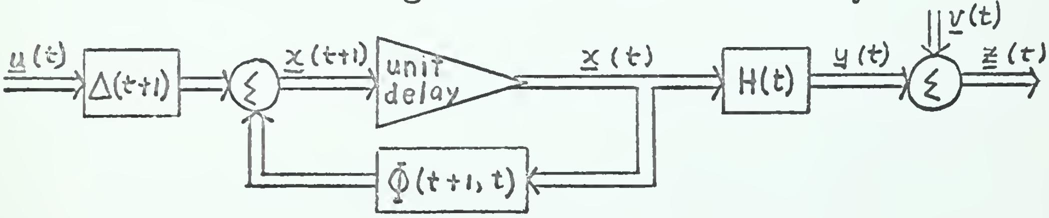

Fig. 1-4 Model for random processes generated by discrete-time linear dynamic systems.

$$E[\underline{v}(t)] = 0 \tag{1.25}$$

$$E[\underline{v}(t)\cdot\underline{v}(t)^T] = \mathrm{cov}[v(t)] = R(t) \tag{1.26}$$

In addition we specify that $\underline{v}(t_1)$ and $\underline{v}(t_2)$ are independent for $t_1 \neq t_2$, i.e:

$$\mathrm{cov}[v(t_1), v(t_2)] = 0 \qquad \text{for } t_1 \neq t_2 \tag{1.27}$$

The output of our model is therefore $\underline{z}(t)$, which contains the observable vector $\underline{y}(t)$ corrupted by additive white noise, $\underline{v}(t)$.

$$\underline{z}(t) = \underline{y}(t) + \underline{v}(t) \tag{1.28}$$

CHAPTER II

THE KALMAN FILTER

2-1 DEFINITION OF THE FILTERING PROBLEM

In Chapter I, a model of a linear dynamic system, excited by white noise, was developed. The purpose of the Kalman filter is to give a best estimate of all states of the system, based on noisy observations of the observable states. Since the system is linear we may write

$$\underline{x}^{*}(t) = \hat{\underline{x}}(t) + G(t)\left[\underline{z}(t) - \hat{\underline{z}}(t)\right] \qquad (2.1)$$

where $\underline{x}^*(t)$ is the best estimate of $\underline{x}(t)$, based on the current observation $\underline{z}(t)$,

$\hat{\underline{x}}(t)$ is the best estimate of $\underline{x}(t)$, based on the previous observation $\underline{z}(t-1)$,

$\hat{\underline{z}}(t)$ is the best estimate of $z(t)$, based on the previous observation $\underline{z}(t-1)$, and,

$G(t)$ is an nxp gain matrix, the magnitude of its elements being indicative of the amount of information carried in $\underline{z}(t)$.

To solve the filtering problem, the filter must therefore determine values for the three unknowns on the right hand side of (2.1), namely $\underline{x}(t)$, $\underline{z}(t)$, and G(t).

2-2 SOLUTION OF THE FILTERING PROBLEM.

Since we assume a complete knowledge of the dynamics of the system, computation of $\hat{\underline{x}}(t)$ and $\hat{\underline{z}}(t)$ is quite simple.

$$\begin{aligned}\hat{\underline{x}}(t) &= E\left[\underline{x}(t) \mid \underline{z}(t-1)\right] \\ &= \Phi(t,t-1)\,\underline{x}^*(t-1) + \Delta(t,t-1)\,E\left[\underline{u}(t-1)\right] \qquad (2.2)\end{aligned}$$

however since $E\left[\underline{u}(t)\right] = 0$ for all t then

$$\hat{\underline{x}}(t) = \Phi(t,t-1)\,\underline{x}^*(t-1) \qquad (2.3)$$

In the model we saw that

$$\begin{aligned}\underline{z}(t) &= \underline{y}(t) + \underline{v}(t) \qquad (1.28)\\ &= H(t)\,\underline{x}(t) + \underline{v}(t)\end{aligned}$$

now

$$\begin{aligned}\hat{\underline{z}}(t) &= E\left[\underline{z}(t) \mid \underline{z}(t-1)\right] \\ &= H(t)\,E\left[\underline{x}(t) \mid \underline{z}(t-1)\right] + E\left[\underline{v}(t)\right] \\ &= H(t)\,\hat{\underline{x}}(t) \qquad (2.4)\end{aligned}$$

since $E\left[\underline{v}(t)\right] = 0$

Having now developed expressions for $\underline{\hat{x}}(t)$ and $\underline{\hat{z}}(t)$, a matrix block diagram of the filter (see Fig. 2-1) can be produced. The only unknown yet to be calculated is the optimal gain matrix G(t). Before approaching this calculation a criterion for optimal must be specified.

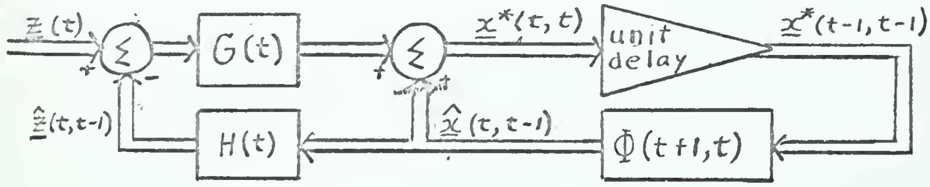

Fig. 2-1 THE OPTIMAL FILTER.

The criterion used is that we wish to find G(t) such that the loss function

$$L = E\left[(\underline{x}(t)-\underline{x}^*(t))^T(\underline{x}(t)-\underline{x}^*(t))\right] \text{ is minimized.} \quad (2.5)$$

That is to say that the sum of the variances of the errors associated with the estimate of the individual states is minimized. Because the errors are gaussian it can be shown (ref. 1) that this criterion will in fact produce an optimal gain matrix.

A number of different derivations for G(t) are available in the literature. For the most part these derivations are mathematically rigorous and somewhat complex. Perhaps the easiest one to follow is a semi-

heuristic approach used by Schmidt, [4]. We wish to minimize the scalar loss function defined by (2.5). Note that

$$E[(\underline{x}-\underline{x}^*)^T(\underline{x}-\underline{x}^*)] = \text{Trace } E[(\underline{x}-\underline{x}^*)(\underline{x}-\underline{x}^*)^T] \qquad (2.6)$$

where the Trace operator denotes the sum of the main diagonal elements, and (t) is left out to avoid unnecessary clutter.

Expanding the covariance matrix in (2.6), using (2.1), we obtain

$$\begin{aligned}E[(\underline{x}-\underline{x}^*)(\underline{x}-\underline{x}^*)^T] &= E\{[(\underline{x}-\hat{\underline{x}})-G(\underline{z}-\hat{\underline{z}})][(\underline{x}-\hat{\underline{x}})-G(\underline{z}-\hat{\underline{z}})]^T\}\\ &= E[(\underline{x}-\hat{\underline{x}})(\underline{x}-\hat{\underline{x}})^T] - E[G(\underline{z}-\hat{\underline{z}})(\underline{x}-\hat{\underline{x}})^T]\\ &\quad -E[(\underline{x}-\hat{\underline{x}})(\underline{z}-\hat{\underline{z}})^T G^T] + E[G(\underline{z}-\hat{\underline{z}})(\underline{z}-\hat{\underline{z}})^T G^T]\end{aligned}$$

but $\underline{z} = H\underline{x} + \underline{v}$, and, $\hat{\underline{z}} = H\hat{\underline{x}}$

Substituting for $\underline{z}$ and $\hat{\underline{z}}$, and noting that $E[(\underline{x}-\hat{\underline{x}})v^T]=0$ we obtain

$$\begin{aligned}E[(\underline{x}-\underline{x}^*)(\underline{x}-\underline{x}^*)^T] &= E[(\underline{x}-\hat{\underline{x}})(\underline{x}-\hat{\underline{x}})^T] - E[GH(\underline{x}-\hat{\underline{x}})(\underline{x}-\hat{\underline{x}})^T]\\ &\quad -E[(\underline{x}-\hat{\underline{x}})(\underline{x}-\hat{\underline{x}})^T HG]\\ &\quad +E\{G(H(\underline{x}-\hat{\underline{x}})(\underline{x}-\hat{\underline{x}})^T H^T + \underline{v}\cdot\underline{v}^T)G^T]\\ &= P - GHP - PH^TG^T + G(HPH^T+R)G^T \qquad (2.7)\end{aligned}$$

where $P = E[(\underline{x}-\hat{\underline{x}})(\underline{x}-\hat{\underline{x}})^T]$ and $R = E[\underline{v}\cdot\underline{v}^T]$.

↑

$P(t,t-1)$

$\hat{P}$?

Conditional covariance

We now wish to find an expression for G such that the trace of (2.7) is a minimum. Since the terms in (2.7) are matrices this could become a very arduous task. Let us consider for a moment that (2.7) is a scalar expression (i.e: the matrices are 1x1 in dimension), thus reducing the right hand side to:

$$P - 2GPH^T + G^2(HPH^T + R)$$

We now differentiate with respect to G and set the resultant to zero obtaining

$$-2PH^T + 2(HPH^T + R)G = 0$$

or $$G = PH^T(HPH^T + R)^{-1} \tag{2.8}$$

It can now be shown that (2.8) will in general provide the optimum gain matrix by letting

$$C = G - PH^T(HPH^T + R)^{-1}$$

or $$G = C + PH^T(HPH^T + R)^{-1} \tag{2.9}$$

Simple substitution of (2.9) for G in (2.7) will reveal that the trace of (2.7) will be a minimum for C=0. Thus (2.8) provides us with the optimum gain

equation.

Combining equations (2.7) and (2.8) results in an expression for the covariance matrix of the error in the filter's estimate of the states of the system;

$$E[(x-x^*)(x-x^*)^T] = P-\hat{P}H^T(HPH^T+R)^{-1}H\hat{P} - \hat{P}H^T[PH^T(HPH^T+R)^{-1}]^T$$

P^* ?

$$+[\hat{P}H^T(H\hat{P}H^T+R)^{-1}][HPH^T+R][PH^T(HPH+R)^{-1}]^T$$

$$= P(t)-\underbrace{\hat{P}(t)H^T(t)\,(H(t)\hat{P}(t)H^T(t)+R(t))^{-1}}_{G}H(t)\hat{P}(t) \quad (2.10)$$

$$= P - GHP = (I-GH)P$$

In order to complete the filtering problem a recursive relation for the conditional covariance matrix, P(t,t-1), must be derived.

Recall that

$$P(t,t-1) = E[(\underline{x}(t)-\hat{\underline{x}}(t,t-1))(\underline{x}(t)-\hat{\underline{x}}(t,t-1))^T]$$

but $\underline{x}(t) = \Phi(t,t-1)\underline{x}(t-1) + \Delta(t,t-1)\underline{u}(t-1)$

and $\hat{\underline{x}}(t,t-1) = \Phi(t,t-1)\underline{x}^*(t-1,t-1)$

$$\therefore\ P(t,t-1) = E\{[\Phi(t,t-1)(\underline{x}(t-1)-\underline{x}^*(t-1,t-1)) + \Delta(t,t-1)\underline{u}(t-1)]\cdot$$

✓

$$[(\underline{x}(t-1)-\underline{x}^*(t-1,t-1))^T\Phi^T(t,t-1) + \underline{u}^T(t-1)\Delta^T(t,t-1)]\}$$

$$= \Phi(t,t-1)E[(\underline{x}(t-1)-\underline{x}^*(t-1,t-1))(\underline{x}(t-1)-\underline{x}^*(t-1,t-1))^T]\cdot$$

$$\Phi^T(t,t-1) + \Delta(t,t-1)E[\underline{u}(t-1)\cdot\underline{u}^T(t-1)]\Delta^T(t,t-1) \quad (2.11)$$

$$P(t,t-1) = \phi\,\underbrace{E[(x-x^*)(x-x^*)^T]}_{\downarrow}\,\phi^T + \Delta\,E[u\,u^T]\,\Delta^T$$

This is P^* for one sample tick;
but it can be expressed as a function of

$x_2 = \Phi$ [illegible]

Δu_k

since the expected values of the cross-product terms are zero.

Define $Q(t) \overset{\Delta}{=} \Delta(t,t-1)E[\underline{u}(t-1)\cdot\underline{u}^T(t-1)]\Delta^T(t,t-1)$ (2.12)

Combining (2.10), (2.11), and (2.12), we obtain

$$\hat{P}(t,t-1) = \underline{\Phi}(t,t-1)\left[\hat{P}(t-1) - P(t-1)H^T(t-1)\left(H(t-1)P(t-1)H^T(t-1)+R(t-1)\right)^{-1}\cdot H(t-1)P(t-1)\right]\underline{\Phi}^T(t,t-1) + Q(t) \tag{2.13}$$

Equation (2.13) is called the variance equation which denotes the covariance of the error between the actual states $\underline{x}(t)$, and the predicted states $\hat{x}(t,t-1)$.

Since (2.13) is recursive, an initial covariance matrix, $P(t_o)$, must be specified, and, since we assume that $\underline{x}(t_o)$ is gaussian, with zero mean, our best estimate of $\underline{x}(t_o)$ is zero. Hence

$$P(t_o) = E[\underline{x}(t_o)\cdot\underline{x}^T(t_o)] \tag{2.14}$$

In determining $P(t_o)$, one will often find that the elements off the main diagonal will be zero, that is to say that the individual initial states are independent of one another. To illustrate this, perhaps a simple example will be helpful. Let us suppose that we are going to make

observations of the position of a particle in motion, and assume that the particle's velocity is constant but unknown. The two states of the system are position and velocity. Any knowledge one might have about one of the initial states will in no way assist in determining the other initial state. Hence the two states are initially independent insofar as the observer is concerned. However, as more observations are made, the two states build up a dependency and the off-diagonal elements in P(t), generated by (2.13), become non-zero.

2-3 SUMMARY OF FILTER EQUATIONS.

For convenience the equations for the optimal filter are grouped below:

$$\left.\begin{array}{lr}
P(t) = \Phi(t,t-1)[P(t-1) - G(t-1)H(t-1)P(t-1)]\Phi^T(t,t-1) + Q(t) & \text{I} \\
G(t) = P(t)H^T(t)[H(t)P(t)H^T(t) + R(t)]^{-1} & \text{II} \\
\underline{\hat{x}}(t,t-1) = \Phi(t,t-1)\,\underline{x}^*(t-1,t-1) & \text{III} \\
\underline{\hat{z}}(t,t-1) = H(t)\,\underline{\hat{x}}(t,t-1) & \text{IV} \\
\underline{x}^*(t,t) = \underline{\hat{x}}(t,t-1) + G(t)[\underline{z}(t) - \underline{\hat{z}}(t,t-1)] & \text{V}
\end{array}\right\}$$

→ Okay; agrees with Schmidt, but not with computer listings.

$$G = PH^T[HPH^T + R]^{-1}$$

CHAPTER III

FILTER DESIGN

3-1 SOME PRELIMINARY CONSIDERATIONS.

The equations for the optimal filter were derived in Chapter II, and are summarized in Section 2-3. Design of the optimal filter consists mainly of writing a suitable computer program to carry out the calculations indicated by the filter equations I through V. The input to the filter is the noisy observation vector, $\underline{z}(t)$, and the output is the best estimate of the system state vector, $\underline{x}^*(t)$.

One of the chief problems in carring out the filter computations is the determination of the inverse of the matrix found in equation II. If this matrix is singular, then the inverse does not exist. One must then resort to the calculation of a pseudo-inverse. The manner in which the pseudo-inverse is determined is shown in [2]. and will not be discussed further here. In either case the time required for inverse computation is relatively high, the time being roughly proportional to the cube of

the dimensionality of the matrix. On present day computers several seconds of computation time may be necessary to determine the inverse of a 4x4 matrix. This being the case, one can easily see that the sampling rate may be adversely affected. In therefore behoves one to make a close study of this matrix to see if computation time can be reduced.

To illustrate, let us assume we are going to track an object in space, receiving position information only. The observable states are range, bearing, and elevation (R, Θ, ϕ), and we wish to determine best estimates of these states along with their derivatives ($\dot{R}$, $\dot{\Theta}$, $\dot{\phi}$). Our state vector would then be set up as follows:

$$\underline{x}(t) = \begin{bmatrix} R \\ \dot{R} \\ \Theta \\ \dot{\Theta} \\ \phi \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} x1 \\ x2 \\ x3 \\ x4 \\ x5 \\ x6 \end{bmatrix}$$

The H matrix would become

$$H(t) = \begin{bmatrix} 100000 \\ 001000 \\ 000010 \end{bmatrix}$$

and hence the matrix (HPH + R) would be 3x3 in dimension.

Recalling that P(t) is symmetrical, computational reduction might be possible if we can assume

$$E\left[\left(\underline{x}_i(t)-\hat{\underline{x}}_i(t,t-1)\right)\left(\underline{x}_{i+2}(t)-\hat{\underline{x}}_{i+2}(t,t-1)\right)\right] \simeq 0$$

$$\text{for } i = 1,4$$

and further that R(t) has non-zero elements along the main diagonal only. Making the above assumptions the P and R matrices would become

$$P(t) = \begin{bmatrix} p11 & p12 & 0 & 0 & 0 & 0 \\ p21 & p22 & 0 & 0 & 0 & 0 \\ 0 & 0 & p33 & p34 & 0 & 0 \\ 0 & 0 & p43 & p44 & 0 & 0 \\ 0 & 0 & 0 & 0 & p55 & p56 \\ 0 & 0 & 0 & 0 & p65 & p66 \end{bmatrix}$$

and $$R(t) = \begin{bmatrix} r11 & 0 & 0 \\ 0 & r22 & 0 \\ 0 & 0 & r33 \end{bmatrix}$$

We would then find that

$$(HPH + R)^{-1} = \begin{bmatrix} \frac{1}{p11 + r11} & 0 & 0 \\ 0 & \frac{1}{p33 + r22} & 0 \\ 0 & 0 & \frac{1}{p55 + r33} \end{bmatrix}$$

since the inverse of a matrix, having non-zero elements along the main diagonal only, is found simply by inverting the diagonal elements.

Suppose now that rate information is also available so that measurements of all six states are made. If, in addition to the above assumptions, we can assume that the cross-variance elements in the P matrix, involving the even subscripted states, are also zero, then

$$(HPH+R)^{-1} = \begin{bmatrix} \begin{bmatrix} a11 & p12 \\ p21 & a22 \end{bmatrix}^{-1} & \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} & \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} \\ \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} & \begin{bmatrix} a33 & p34 \\ p43 & a44 \end{bmatrix}^{-1} & \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} \\ \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} & \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} & \begin{bmatrix} a55 & p56 \\ p65 & a66 \end{bmatrix}^{-1} \end{bmatrix}$$

where $aii = pii + rii, \; i = 1,6$

thus reducing computational time by at least $6^3/3x2^3$ or 9 times.

A further reduction might be realized in the given example by the use of three filters doing 2x2 matrix manipulations as opposed to one filter computing at the 6x6 level.

Another consideration is the time lag between input and output. In real time situations this could be of the

utmost importance. A glance at the filter equations, I-V, indicates that if all five equations are computed after the input, $\underline{z}(t)$, has been received, then a considerable time lag could ensue before the output is generated. On the other hand, if the transition matrix, $\Phi(t,t-1)$, is known at time t-1, then equations I, II, III, and IV can be computed prior to receiving the input. The time delay in this case would only involve the time taken to compute V, which might be in the order of micro-seconds.

3-2 FILTER FOR A STATIONARY PROCESS.

It is to be noted that equations I and II do not involve the observations, $\underline{z}(t)$. If the process, which we are trying to observe, is stationary, i.e; Φ, H, Q, and R are constant matrices, then it will be found that the optimal gain matrix, G, will stabilize to a constant value. This matrix could then be precomputed (prior to any observations), and the filter would be reduced to the relatively simple calculations indicated by III, IV, and V.

A digital program, which computes the optimum gain matrix for a stationary dynamic process, has been written in FORTRAN, and is found in Appendix II. It

is completely general and can handle any system with up to twelve states. It is written as a subroutine to eliminate possible conflict in the naming of variables. Essentially the subroutine carries out the iterations indicated by I and II until such time that no element of the gain matrix changes from its previous value by an amount more than .00001. If higher accuracy is desired, this constant can be easily changed to the degree of accuracy required. Further description on the usage of this program is found in the Appendix.

3-3 THE GENERAL FILTER.

In the general case, non-stationarity is assumed. Appendix III contains two programs, the first of which performs the computations indicated by the five filter equations after each observation. The second programme allows for the case when the transition matrix is known before an observation is made and hence reduces the time lag (previously discussed) between out put and input. Further discription of the usage of these programs is contained in the Appendix.

CHAPTER IV

ILLUSTRATIVE EXAMPLES

This section of the paper will deal with two illustrative examples. The aim of the first example is primarily tutorial. A thorough discussion of the model will be given, followed by the design of an optimal filter. The second example will deal with the track smoothing of an anti-ship missile. A mathematical model of the missile in flight will be set up and a filter designed for optimum track smoothing.

EXAMPLE I:

Given the transfer function for a low pass filter in Fig. 1, a) determine all mathematical model parameters, and, b) design an optimal filter which will give a best estimate of the states in the filter. Assume that the excitation at the input (u(t)), and the additive noise (v(t)) at the output are gaussian and stationary.

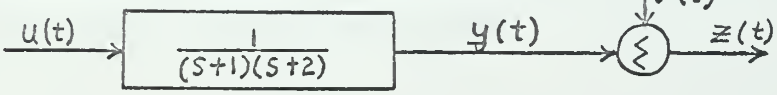

Fig. 4-1 Low pass filter of Example I.

Our first step is to convert Fig. 4-1 to a more convenient form for analysis as shown in Fig. 4-2.

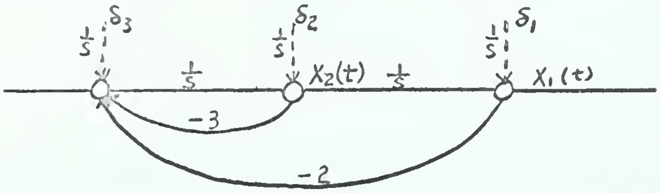

Fig. 4-2 Signal flow graph for Example I

From Chapter I, equations (1.8) through (1.12), we see that $\dot{x}(t) = Fx(t) + Du(t)$ (4.1)

or

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ u(t) \end{bmatrix}$$

We wish now to find the solution to (4.1) in the form given by (1.15). To do so we must assume that u(t) is piece-wise constant.

The Φ and Δ matrics will be obtained by using signal flow techniques and applying unit impulses at selected locations in the diagram as illustrated in Fig. 4-2.

From Fig. 4-2 we see that

$$\Phi(s) = \begin{bmatrix} \frac{x_1(s)}{\delta_1} & \frac{x_1(s)}{\delta_2} \\ \frac{x_2(s)}{\delta_1} & \frac{x_2(s)}{\delta_2} \end{bmatrix}$$

and

$$\Delta(s) = \begin{bmatrix} 0 & \dfrac{x_1(s)}{\delta_3} \\ 0 & \dfrac{x_2(s)}{\delta_3} \end{bmatrix}$$

$$\phi_{11}(s) = \frac{x_1(s)}{\delta_1} = \frac{\frac{1}{s}\left(1+\frac{3}{s}\right)}{1+\frac{3}{s}+\frac{2}{s^2}} = \frac{s+3}{s^2+3s+2}$$

$$\phi_{12}(s) = \frac{x_1(s)}{\delta_2} = \frac{\frac{1}{s^2}}{1+\frac{3}{s}+\frac{2}{s^2}} = \frac{1}{s^2+3s+2}$$

$$\phi_{21}(s) = \frac{x_2(s)}{\delta_1} = \frac{-\frac{2}{s^2}}{1+\frac{3}{s}+\frac{2}{s^2}} = \frac{-2}{s^2+3s+2}$$

$$\phi_{22}(s) = \frac{x_2(s)}{\delta_2} = \frac{\frac{1}{s}}{1+\frac{3}{s}+\frac{2}{s^2}} = \frac{s}{s^2+3s+2}$$

$$\Delta_{12}(s) = \frac{x_1(s)}{\delta_3} = \frac{\frac{1}{s^3}}{1+\frac{3}{s}+\frac{2}{s^2}} = \frac{1}{s(s^2+3s+2)}$$

$$\Delta_{22}(s) = \frac{x_2(s)}{\delta_3} = \frac{\frac{1}{s^2}}{1+\frac{3}{s}+\frac{2}{s^2}} = \frac{1}{s^2+3s+2}$$

Taking the inverse of the above and letting the sample interval be T we find

$$\underline{\Phi}(t+T,t) = \begin{bmatrix} 2e^{-T}-e^{-2T} & e^{-T}-e^{-2T} \\ 2e^{-2T}-2e^{-T} & 2e^{-2T}-e^{-T} \end{bmatrix} \qquad (4.2)$$

and,

$$\Delta(t+T,t) = \begin{bmatrix} 0 & \frac{1}{2} - e^{-T} + \frac{1}{2}e^{-2T} \\ 0 & e^{-T} - e^{-2T} \end{bmatrix} \qquad (4.3)$$

The solution is now in the form of (1.15)

$$\underline{x}(t+T) = \Phi(t+T,t)\,\underline{x}(t) + \Delta(t+T,t)\,\underline{u}(t) \qquad (1.15)$$

and the mathematical model is shown in Fig. 4-3,

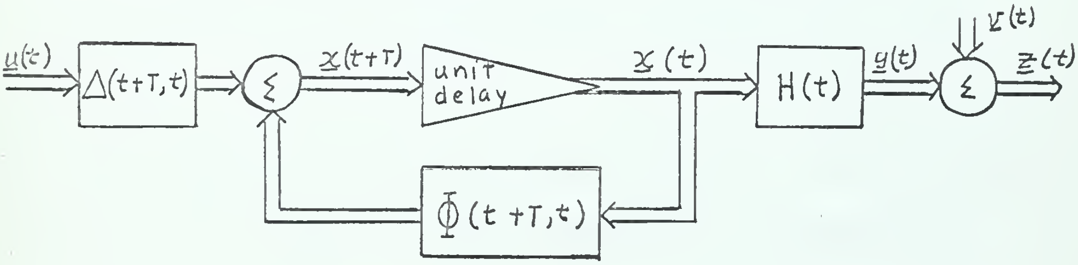

Figure 4-3 Mathematical model for Example I

where H(t) is obviously $\begin{bmatrix}1 & 0\end{bmatrix}$, since only one state, x_1, is observed.

b) Design of optimal filter

We recall that optimum filtering is based upon a knowledge of the statistics of u(t) and v(t), and therefore we assume that $E\left[\underline{u}(t)\cdot\underline{u}^T(t)\right]$ and $E\left[\underline{v}(t)\cdot\underline{v}^T(t)\right]$ are part of the problem statement. We now calculate the covariance matrix

$$Q(t+T) = \Delta(t+T,t)\,E\left[\underline{u}(t)\cdot\underline{u}^T(t)\right]\Delta^T(t+T,t) \qquad (4.4)$$

$$\text{and} \quad R(t) = E\left[\underline{\nu}(t) \cdot \nu^T(t)\right] = r_{11} \qquad (4.5)$$

The only remaining task is to select an initial covariance matrix, $P(t_o)$, for our best estimates of the initial states of the model. The selection will depend to a great extent on a good knowledge of the problem. In this instance the best selection would probably be the main diagonal of $Q(t)$ (previously calculated) with the off-diagonal terms set to zero. The off-diagonal terms are zero initially since knowledge of $x_1(t_o)$ (at the first measurement) will in no way provide any information about $x_2(t_o)$ ie $x_1(t_o)$ and $x_2(t_o)$ are uncorrelated insofar as the observer is concerned.

Since the system under study is stationary, the optimum gain matrix may be pre-determined. This entails the use of SUBROUTINE CONFIL in Appendix II. A sample period of 0.1 secs. is used. Using (4.3) and (4.4), we compute the covariance of the states, $Q(t)$. The element q_{11} in $Q(t)$ is a measure of the expected signal power. By making r_{11} (measurement noise power) equal to various multiples of q_{11} (including

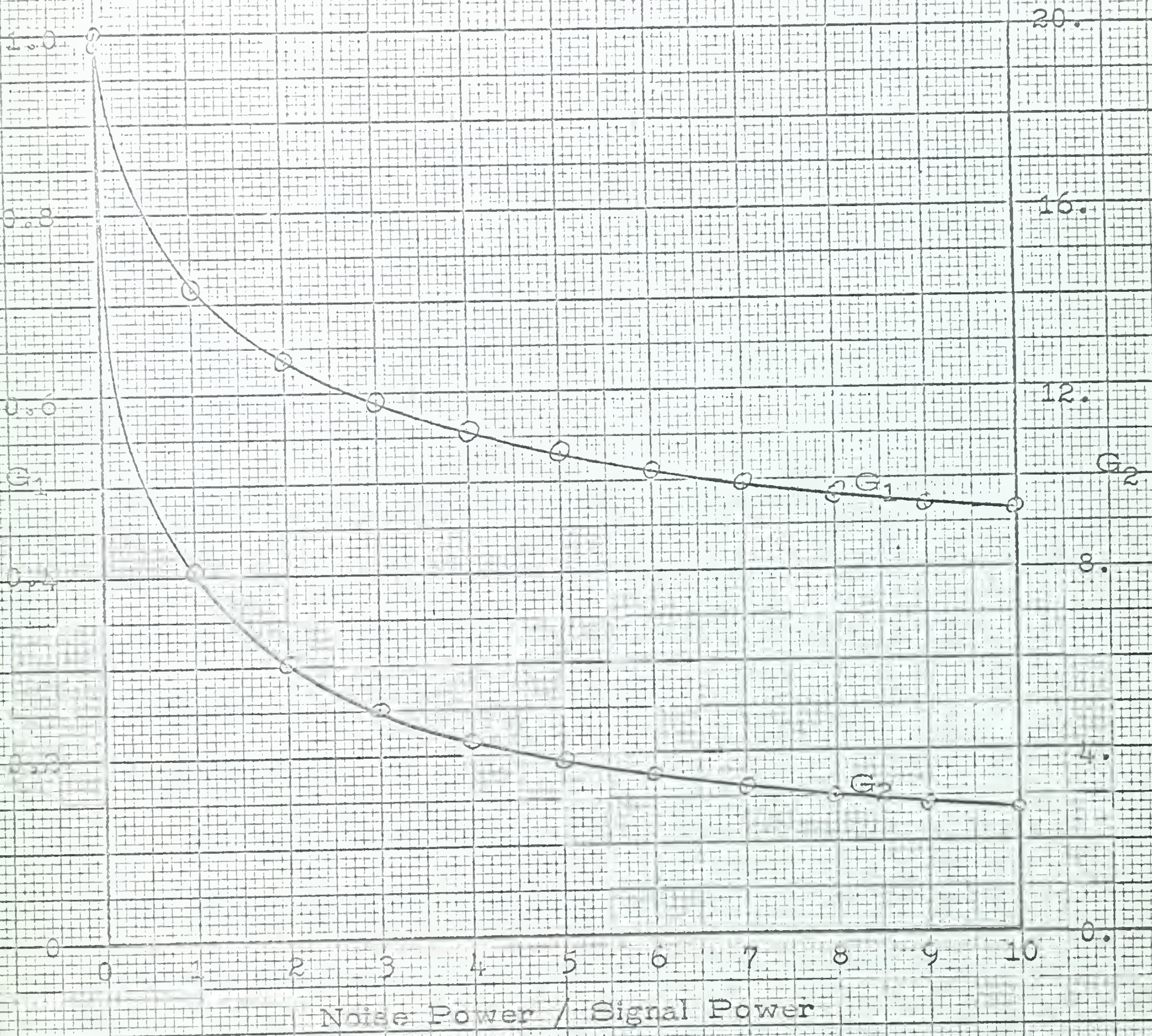

Fig. 4-4 Curves showing how the optimum gain matrix elements vary as a function of noise-to-signal power for Example I.

K+E 10 X 10 TO ½ INCH 46 1470
7½ X 10 INCHES MADE IN U.S.A.
KEUFFEL & ESSER CO.

zero) we are able to study the behaviour of the gain matrix as a function of noise-to-signal power.

The curues in Figure 4-4 indicate how the optimum gain matrix elements behave as noise-to-signal power is increased. With N/S = 0 we see that G1 is equal to unity. We expect this since there is no measurement noise and hence the best estimate of the observable state is the measurement z(t) itself. However as the noise power is increased the gain element falls off and the filter starts to rely more on the predicted value and less on the observed value. The matrix element G2 also falls off in a similar fashion, with $x^*_2(t)$ becoming less dependent on z(t) as the relative noise power increases.

EXAMPLE II

Problem Statement - It is known that the enemy's main anti-ship weapon is an air-launched missile which is normally launched at a distance of 250 to 300 miles from target. After launch, it climbs to an altitude of 40,000 ft., attains a speed of approximately 1000 mph, and maintains this speed by use of a sustainer motor. When

within 25 miles of the expected location of target a search device is switched on which pinpoints the target and enables the missile to guide itself to target. A typical friendly ship at sea is fitted with a mono-pulse search radar system with a scan rate of 10 scans/min. The ship is fitted with a digital computer and target information is available to the computer. It is known that the probability distribution function of the accumulated error (in both x and y) is approximately normal with zero mean and 2 mile standard deviation.

Data accumulated on similar missiles indicate that, due to erratic thrust developed by the missile motor and average atmospheric turbulence, the velocity of the missile varies in a random fashion (approximately gaussian). The standard deviation of this randomness is about 2% of mean velocity.

?

Design a filter which will determine best estimates of the missile's position and velocity. Assume that attack is equi-likely from all directions.

SOLUTION

Our first step is to set up a mathematical model

which describes the dynamics of the missile, in the form

$$\dot{\underline{x}}(t) = F(t)\,\underline{x}(t) + D(t)\,\underline{u}(t)$$

Considering only the component in the x direction we obtain

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ u_x(t) \end{bmatrix} \qquad (4.6)$$

$\underline{u}$ is vector

where x_1 is the position on the x axis

x_2 is the velocity component in the x direction

A similar vector equation would describe the dynamics in the y direction.

As we have seen earlier the solution to the above equation is

$$\underline{x}(t) = \Phi(t,t_1)\,\underline{x}(t_1) + \Delta(t,t_1)\,\underline{u}_x(t) \qquad (4.7)$$

assuming $u_x(t)$ is piecewise constant.

We now must determine $\Phi(t_2,t_1)$.

From (4.4) we obtain the model shown in Fig 3.5., a simple $1/s^2$ plant.

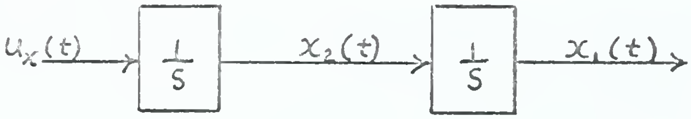

Fig 3.5 Model for missle in Example II.

The parameters Φ and Δ are

$$\Phi(t_2, t_1) = \begin{bmatrix} 1 & t_2 - t_1 \\ 0 & 1 \end{bmatrix}$$

$$\Delta(t_2, t_1) = \begin{bmatrix} 0 & \frac{(t_2 - t_1)^2}{2} \\ 0 & t_2 - t_1 \end{bmatrix}$$

Since we intend to sample at a rate of 10 times/ min. (scan rate of radar) then $t_2 - t_1 = 6$ seconds.

or $T = \frac{1}{600}$ hrs.

$$\Phi(t, t-T) = \begin{bmatrix} 1 & \frac{1}{600} \\ 0 & 1 \end{bmatrix}$$

and

$$\Delta(t, t-T) = \begin{bmatrix} 0 & \frac{1}{2}\left(\frac{1}{600}\right)^2 \\ 0 & \frac{1}{600} \end{bmatrix}$$

Our next step is to find Q(t)

no puf x here

OK for I–V equations

$$\begin{bmatrix} U_{x_1}^{2} & U_{x_1}U_{x_2} \\ U_{x_1}U_{x_2} & U_{x_2}^{2} \end{bmatrix}$$

$$\underline{Q(t)} = \Delta(t, t-T)\, \underbrace{E[u(t-T) \cdot u^{T}(t-T)]}\, \Delta^{T}(t, t-T)$$

$$= \begin{bmatrix} 0 & \frac{T^2}{2} \\ 0 & T \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & E[u_x^2] \end{bmatrix} \begin{bmatrix} 0 & 0 \\ \frac{T^2}{2} & T \end{bmatrix}$$

because u is a vector

$$= \begin{bmatrix} 0 & \frac{T^2}{2} E[u_x^2] \\ 0 & T\, E[u_x^2] \end{bmatrix} \begin{bmatrix} 0 & 0 \\ \frac{T^2}{2} & T \end{bmatrix}$$

$$= \begin{bmatrix} \frac{T^4}{4} E[u_x^2] & \frac{T^3}{2} E[u_x^2] \\ \frac{T^3}{2} E[u_x^2] & T^2 \cdot E[u_x^2] \end{bmatrix}$$ ✓

The problem statement specified however that

$$E[(x_2 - \bar{x}_2)^2] = (0.02\, \bar{V}_x)^2 = \boxed{q_{22}}$$

? why $= q_{22}$??

$q_{22} = T^2 E[u_x^2]$

$$\therefore \quad E[(u_x^2)] = \frac{(0.02\, \bar{V}_x)^2}{T^2}$$

$$\therefore \quad Q(t) = (0.02\, \bar{V}_x)^2 \begin{bmatrix} \frac{T^2}{4} & \frac{T}{2} \\ \frac{T}{2} & 1 \end{bmatrix}$$ ✓

$u = \begin{bmatrix} 0 \\ u_x \end{bmatrix}$

variance $= \sigma^2$

$\sigma = 0.02\, \bar{V}_x$

U_{x_1}

U_{x_2}

$u\, u^T$

$\begin{bmatrix} 0 \\ u_x \end{bmatrix} \begin{bmatrix} 0 & u_x \end{bmatrix}$

$\begin{bmatrix} 0 & 0 \\ 0 & E(u_x \end{bmatrix}$

To complete the model we require values for H(t) and R(t).

$\sigma = 2$

variance $= \sigma^2 = 4$

From the problem statement

$$R(t) = r_{11} = 2^2 = 4.0$$

and since we are observing position only

$$H(t) = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

Summarizing our model we have (Fig. 4.6)

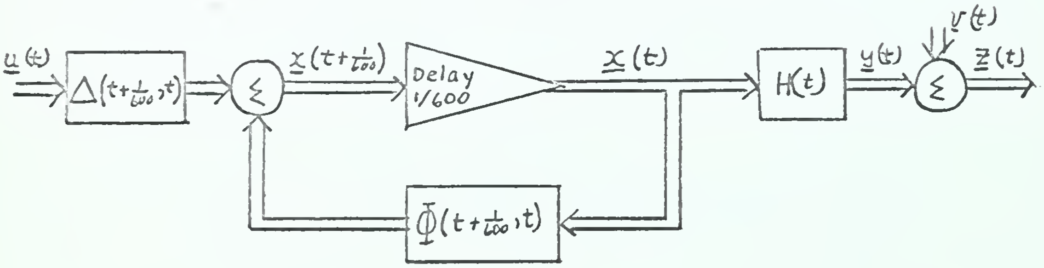

Fig 4.6 Model for Missle of Example II

We are now ready to set up the filter. One requirement is the initial covariance matrix P(o). Since the missile can approach us from any direction with equal probability our best estimate for the inital states is zero for both position and velocity, ie

$$\begin{bmatrix} \hat{x}_1(0) \\ \hat{x}_2(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

We therefore find that

$P(o) = E\left[(x-\hat{x})(x-\hat{x})^T\right]$

but $\hat{x} = 0$

$$P(0) = E[\underline{x}(0)\cdot x^T(0)]$$

To determine this we must have a knowledge about the detection capability of the radar. A standard search radar looking for an object at 40,000 ft. elevation may have an initial detection density function, as shown in Figure 4.7, in all directions in the x-y plane.

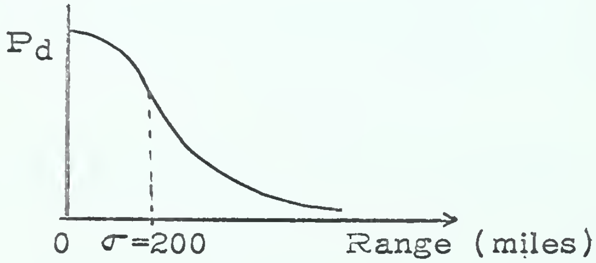

FIG. 4.7 Probability density function for inital detection

→ why?

Since the missile may approach from any direction let us assume that the standard deviation when considering the x direction only is 100 miles. Similarly let us assume the velocity component on the x axis has a standard deviation of 500 mph. We then find that

$$P(0) = \begin{bmatrix} (100)^2 & 0 \\ 0 & (500)^2 \end{bmatrix}$$

Our last remaining task is to select a value for $\overline{V}_x^2(0)$ to provide numbers for the Q(o) matrix. Our best initial estimate of this quantity would be the p_{22} element in P(o).

Hence

$$\overline{V}_x^2(0) = (500)^2$$

why not 1000?

Subsequent values of $\overline{V}_x(t)$ could be calculated by using $\hat{x}_2(t)$.

We are now prepared to write a program for the optimal filter.

Since the transition matrix is a constant but Q is variable (being a function of V_x) the 2nd program of Appendix III would apply.

We now summarize taking into account the y component of direction. The flow chart for the filter calculations is shown in Appendix III.

n (number of states) = 4

p (number of observables) = 2

T (sample interval) = 1/600 hrs.

$$\underline{x}(t) = \begin{bmatrix} x(t) \\ \dot{x}(t) \\ y(t) \\ \dot{y}(t) \end{bmatrix} = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix}$$

$$\underline{y}(t) = \begin{bmatrix} x_1(t) \\ x_3(t) \end{bmatrix}$$

$$\underline{z}(t) = \begin{bmatrix} x_1(t) + x(t) \\ x_3(t) + y(t) \end{bmatrix}$$

why? ??

$$H(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\underline{\Phi}(t+T,t) = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$q_{22}(o) = q_{44}(o) = (.02 \times 500)^2 = 100$$

$$Q(t) = \begin{bmatrix} \frac{T^2}{4} q_{22} & \frac{T}{2} q_{22} & 0 & 0 \\ \frac{T}{2} q_{22} & q_{22} & 0 & 0 \\ 0 & 0 & \frac{T^2}{4} q_{44} & \frac{T}{2} q_{44} \\ 0 & 0 & \frac{T}{2} q_{44} & q_{44} \end{bmatrix}$$

$$R(t) = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}$$

$$P(o) = \begin{bmatrix} (100)^2 & 0 & 0 & 0 \\ 0 & (500)^2 & 0 & 0 \\ 0 & 0 & (100)^2 & 0 \\ 0 & 0 & 0 & (500)^2 \end{bmatrix}$$

$$G(o) = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$\hat{x}(0) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

✓

$$\hat{z}(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

BIBLIOGRAPHY

1. Kalman, R. E.; "A New Approach to Linear Filtering and Prediction Problems"; Journal of Basic Engineering, March, 1960.

2. Kalman, R. E.; "New Methods and Results in Linear Prediction and Filtering Theory"; Rias Technical Report 61-1, 1961.

3. Magill, T. D.; "Optimal Adaptive Estimation of Sampled Stochastie Processes"; Stanford Electronics Laboratories Technical Report No. 6302-2, December 1963.

4. Schmidt, S. F.; "The Application of State Space Methods to Navigation Problems"; Philco Technical Report No. 4, July 1964.

APPENDIX I

LINEAR DIFFERENTIAL EQUATIONS

Let us consider the solution of a 1st order differential equation by the use of an integrating factor (p), to make an exact differential.

Given:

$$\frac{dx}{dt} + ax = u \tag{1}$$

Find:

$$x(t) = f(x_o, u, a, t)$$

Solution:

Multiply (1) by p(t) and attempt to make the resulting equation an exact differential. We have

$$p\frac{dx}{dt} + pax = pu$$

or (2)

$$\frac{d}{dt}(px) - \frac{dp}{dt}x + apx = pu$$

and

$$\frac{d}{dt}(px) = \left(\frac{dp}{dt} - ap\right) x + pu \tag{3}$$

Considering the homogeneous part of the problem (i.e., $u = 0$), we can make (3) an exact differential by setting

$$\frac{dp}{dt} - ap = 0 \qquad (4)$$

We may guess at a solution for (4) as

$$p = p_o e^{at} \qquad (5)$$

and by substitution in (4), verify that it is a solution. Applying this to (3) gives

$$\frac{d}{dt}\left(p_o e^{at} x\right) = (0)\, x + p_o e^{at} u \qquad (6)$$

Integrating

$$p_o \left[e^{at} x - x_o = \int_o^t e^{a\tau} u(\tau)\, d\tau \right] \qquad (7)$$

Multiplying by e^{-at} gives

$$x = e^{-at} x_o + e^{-at} \int_o^t e^{a\tau} u(\tau)\, d\tau \qquad (8)$$

or

$$x = e^{-at} x_o + \int_o^t e^{-a(t-\tau)} u(\tau)\, d\tau \qquad (9)$$

The first part of (9) represents the homogeneous solution while the latter part

$$x_p = \underline{\int_o^t e^{-a(t-\tau)} u(\tau)\, d\tau}$$

represents the particular solution or convolution integral.

Now let us consider a set of n of these 1st order differential equations

$$\frac{dx_i}{dt} = f_i(x_j, u) \qquad i = 1, \ldots . 7$$

or

$$\dot{\underline{x}} = F\underline{x} + D\underline{u} \tag{11}$$

where $\underline{x}$ and $\underline{u}$ are 1 x n column vectors and F and D are n x n matrices. Multiplying (11) by the integrating factor $\bar{p}(t)$, (a row vector) we obtain, after some manipulation

$$\frac{d}{dt}\left(\bar{p}\, \underline{x} \right) = \overbrace{\left(\frac{d\bar{p}}{dt} + \bar{p}F \right)}^{=0} \underline{x} + \bar{p}D\underline{u} \tag{12}$$

As before we assume a solution for the adjoint variable, $\bar{p}(t)$ as

$$\bar{p} = \bar{p}_o e^{-Ft}$$

Substituting into (12) and integrating gives

$$\bar{p}_o \left[e^{-Ft} \underline{x} - \underline{x}_o = \int_o^t e^{-F\tau} D\underline{u}(\tau)\, d\tau \right] \tag{13}$$

Multiplying both sides by e^{Ft} gives

$$\underline{x} = e^{Ft}\, \underline{x}_o + \int_o^t e^{F(t-\tau)}\, D\underline{u}\,(\tau)\, d\tau \tag{14}$$

Often the above is expressed in the form

$$\underline{x}(t_1) = \underline{\Phi}(t_1, t_o)\, \underline{x}(t_o) + \Delta(t_1, t_o)\, u(t_o) \tag{15}$$

where $\underline{\Phi}$ is the transition matrix or fundamental matrix, and Δ represents the distribution matrix with $u(\tau)$ held constant at its value $u(t_o)$.

APPENDIX II

THE STATIONARY FILTER

PURPOSE:

The attached program can be used to determine the stabilized optimum gain matrix for the estimation of the states of a stationary process.

USAGE:

1. Calling Sequence

CALL CONFIL (P, Q, R, TR, H, KN, KP, KER, G)

2. Arguments

a.	P	initial covariance matrix of system states - dimension (12 x 12)
b.	Q	covariance matrix of states due to gaussian excitation - dimension (12 x 12)
c.	R	covariance matrix of measurement noise - dimension (12 x 12)
d.	TR	transition matrix of process - dimension (12 x 12)
e.	H	matrix which defines the observable states - dimension (12 x 12)
f.	KN	number of states in the system - dimension (scaler)

g. KP - number of observable states - dimension (scaler)

h. KER - error indication (= 2 implies the inverse of a matrix could not be obtained) - dimension (1)

i. G - optimal gain matrix - dimension (12 x 12)

3. Accuracy: see Chapter III

4. Equipment Configuration: CDC 1604 with FORTRAN 60.

5. Cautions to User:

a. All arguments in the main program must be dimensioned the same as those in CONFIL.

b. The main program should contain an ERROR TEST (see 2h.)

6. Flow chart showing Typical Usage (see Fig. 11-1).

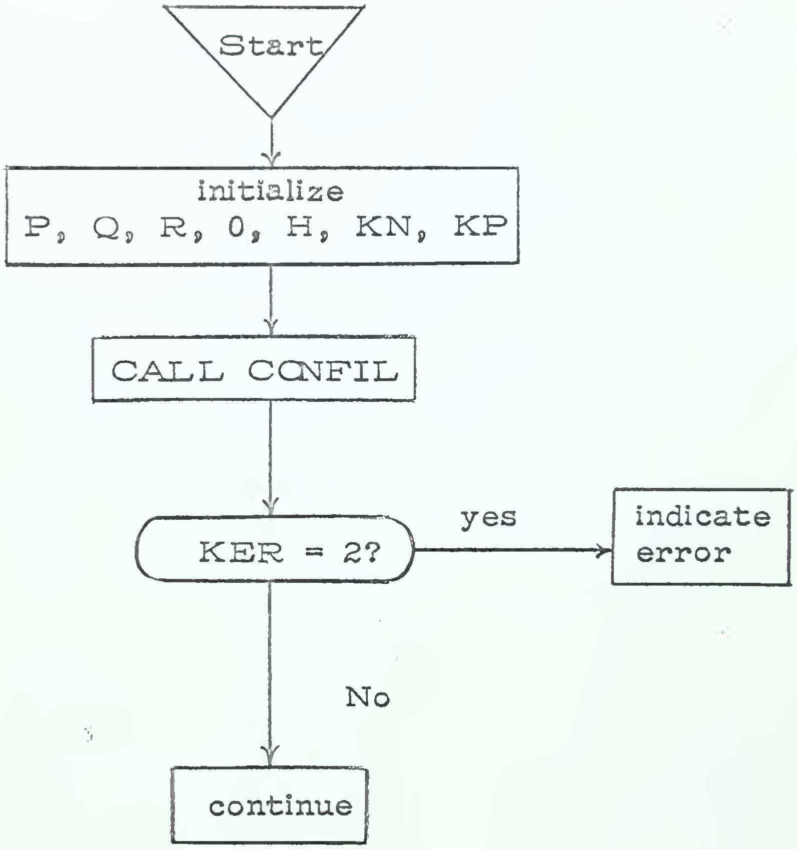

Fig. 11-1 Flow diagram showing typical usage of CONFIL

SUBROUTINE CONFIL

```
      SUBROUTINE CONFIL(P,Q,R,TR,H,KN,KP,KER,G)
C     THIS SUBROUTINE COMPUTES THE STABILIZED GAIN MATRIX TO BE USED
C     IN A FILTER FOR THE OPTIMAL ESTIMATION OF THE STATES OF A
C     STATIONARY SAMPLED-DATA LINEAR DYNAMIC PROCESS
     0DIMENSION P(12,12),Q(12,12),H(12,12),TR(12,12),TS(12,12),DEL(12,12
     1),G(12,12),R(12,12),GT(12,12),HT(12,12),TEMP(12,12),TEMP2(12,12)
     2,TEMP1(12,12),KER(1)
      DO 102 I=1,KN
      DO 102 J=1,KP
  102 GT(I,J)=0.
      CALL TRANS (H,KP,KN,HT)

C     BEGIN ITERATIVE LOOP TO CALCULATE G(T)

  104 CALL PROD (P,HT,KN,KN,KP,TEMP)
      CALL PROD (H,TEMP,KP,KN,KP,TEMP2)
      CALL ADD (TEMP2,R,KP,KP,TEMP2)
      CALL RECIP (KP,.00001,TEMP2,TEMP1,KER)
      IF (KER-2) 101,110,101
  101 CALL PROD(TEMP,TEMP1,KN,KP,KP,G)

C     BEGIN TEST FOR STABILIZATION OF G(T)

      DO 106 I=1,KN
      DO 106 J=1,KP
      X= ABSF (1. -(GT(I,J)+.000001)/(G(I,J)+.000001))
      IF(.00001-X) 105,105,106
  105 GO TO 107
  106 CONTINUE
```

TEMP= [3X3] [⋮]

```
C     END STABILIZATION TEST

      GO TO 110

C     G(T) NOT STABILIZED CONTINUE ITERATION

  107 CALL PROD (TR,G,KN,KN,KP,DEL)
      CALL PROD (DEL,H,KN,KP,KN,TEMP)
      DO 108 I=1,KN
      DO 108 J=1,KN
  108 TEMP(I,J)= -TEMP(I,J)
      CALL ADD (TR,TEMP,KN,KN,TS)
      CALL TRANS(TS,KN,KN,TEMP)
      CALL PROD (P,TEMP,KN,KN,KN,TEMP2)
      CALL PROD (TS,TEMP2,KN,KN,KN,P)
      CALL ADD(P,Q,KN,KN,P)
      DO 109 I=1,KN
      DO 109 J=1,KP
  109 GT(I,J)=G(I,J)
      GO TO 104

C     LOOP AGAIN

  110 CONTINUE
      END
      SUBROUTINE PROD (A,B,N,M,L,C)
      DIMENSION A(12,12),B(12,12),C(12,12)
      DO 151 I=1,N
      DO 151 J=1,L
      C(I,J) =0.
      DO 151 K = 1,M
  151 C(I,J) = C(I,J) + A(I,K)*B(K,J)
      END
      SUBROUTINE ADD (A,B,N,M,C)
      DIMENSION A(12,12),B(12,12),C(12,12)
```

```
      DO 152 I=1,N
      DO 152 J=1,M
  152 C(I,J) = A(I,J) + B(I,J)
      END
      SUBROUTINE TRANS(A,N,M,C)
      DIMENSION A(12,12),C(12,12)
      DO 153 I = 1,N
      DO 153 J=1,M
  153 C(J,I) = A(I,J)
      END
      SUBROUTINE RECIP(N,EP,A,X,KER)
      DIMENSION A(12,12),X(12,12)
      DO 1 I=1,N
      DO 1 J=1,N
    1 X(I,J)=0.0
      DO 2 K=1,N
    2 X(K,K)=1.0
   10 DO 34 L=1,N
      KP=0
      Z=0.0
      DO 12 K=L,N
      IF(Z-ABSF(A(K,L)))11,12,12
   11 Z=ABSF(A(K,L))
      KP = K
   12 CONTINUE
      IF(L-KP)13,20,20
   13 DO 14 J=L,N
      Z=A(L,J)
      A(L,J)=A(KP,J)
   14 A(KP,J)=Z
      DO 15 J=1,N
      Z=X(L,J)
      X(L,J)=X(KP,J)
   15 X(KP,J)=Z
   20 IF(ABSF(A(L,L))-EP)50,50,30
```

APPENDIX II   II-   SUBROUTINE CONFIL (CONTINUED)

```
30 IF(L-N)31,34,34
31 LP1=L+1
   DO 36 K=LP1,N
   IF(A(K,L))32,36,32
32 RATIO=A(K,L)/A(L,L)
   DO 33 J=LP1,N
33 A(K,J)=A(K,J)-RATIO*A(L,J)
   DO 35 J=1,N
35 X(K,J)=X(K,J)-RATIO*X(L,J)
36 CONTINUE
34 CONTINUE
40 DO 43 I=1,N
   II=N+1-I
   DO 43 J=1,N
   S=0.0
   IF(II-N)41,43,43
41 IIP1=II+1
   DO 42 K=IIP1,N
42 S=S+A(II,K)*X(K,J)
43 X(II,J)=(X(II,J)-S)/A(II,II)
   KER=1
   RETURN
50 KER=2
   END
   END
```

APPENDIX II II- SUBROUTINE CONFIL (CONTINUED)

APPENDIX III

GENERAL FILTERS

PREAMBLE: Two programmes, in subroutine form, are contained in this appendix. The first programme carries out all filter computations after the input has been received at each sample instant. The time lag between output and input will therefore depend on the time taken for these computations. The second programme is designed for use when certain parameters are known (or at least a very good estimate of these parameters can be made) prior to receiving the input at each sample instant. A prior knowledge of these parameters, namely TR(t,t-1), H(t), R(t), Q(t) (defined below), enables a considerable reduction of the above-mentioned time lag between output and input. This is achieved by performing most of the filter computations prior to receiving the input.

A. SUBROUTINE BESTX

PURPOSE: This subroutine will provide an optimum estimate of the state vector for any sampled-data

linear dynamic process (with twelve or less states) if both the process random excitation and corruptive measurement noise vectors have gaussian distributions.

USAGE:

1. Calling Sequence

CALL BESTX (P, Q, R, TR, H, KN, KP, KER, G, XP, Z, X)

2. Arguments

a. P - initial covariance matrix of system states -
dimension (12 x 12)

b. Q - covariance matrix of states due to gaussian excitation -
dimension (12 x 12)

c. R - covariance matrix of measurement noise -
dimension (12 x 12)

d. TR - transition matrix of process -
dimension (12 x 12)

e. H - matrix which defines the observable states -
dimension (12 x 12)

f. KN - number of states in the system -
dimension (scaler)

g. KP - number of observables states -
dimension (scaler)

h. KER - error indication (=2 implies the inverse of a matrix could not be obtained) -
dimension (1)

i. G - optimal gain matrix -
dimension (12 x 12)

j. XP - predicted estimate of process state vector - dimension

k. Z - observation vector - dimension (12 x 12)

l. X - optimal estimate of process state vector (generated by filter) - dimension (12 x 12)

3. Equipment Configuration: CDC 1604 with FORTRAN 60.

4. Cautions to User:

a. All arguments in the main program must be dimensioned the same as those in BESTX

b. The main program should contain an ERROR TEST (see 2h.)

c. See attached flow chart depicting typical usage.

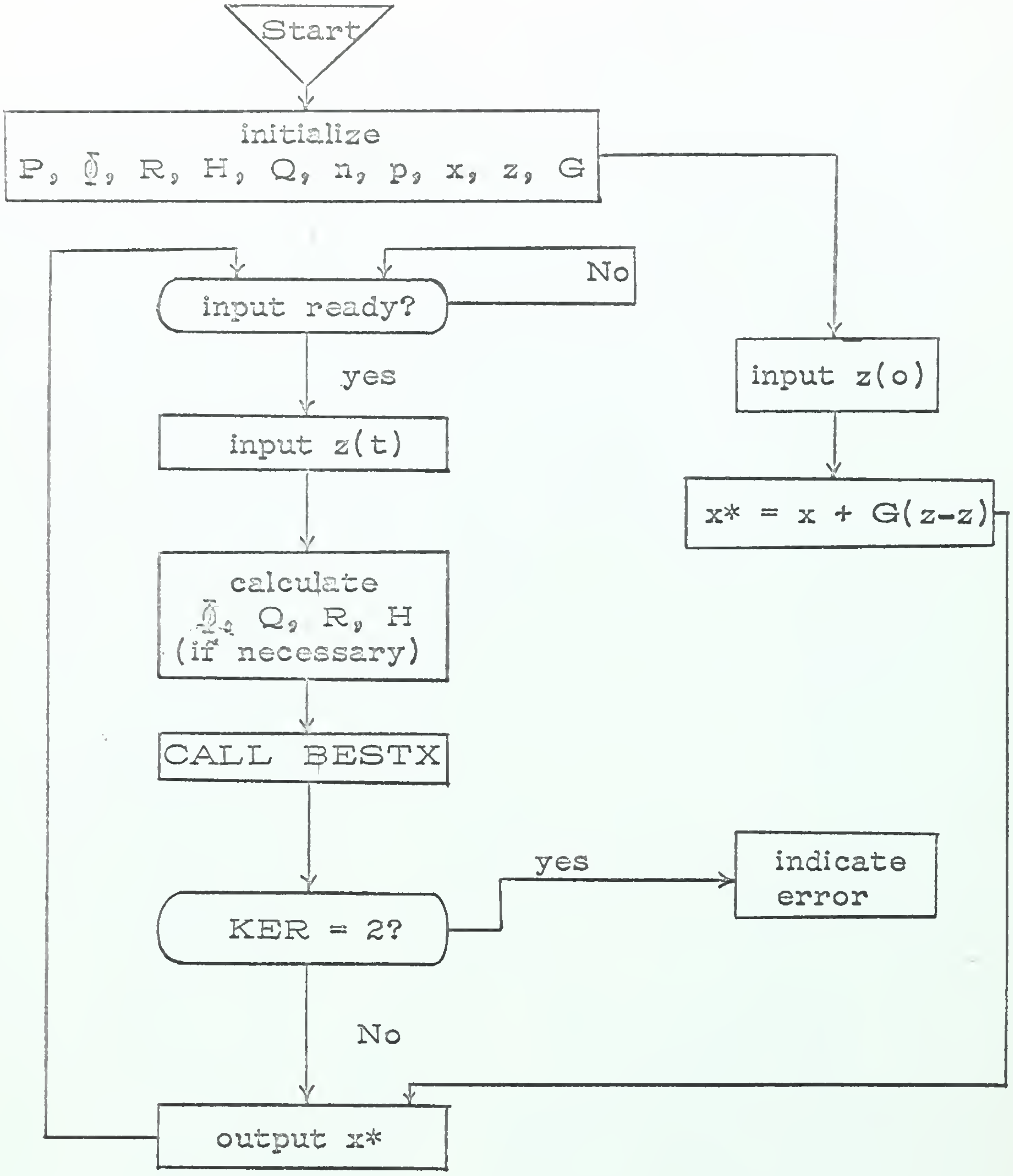

Fig. 111-1 Flow diagram showing typical usage of BESTX

SUBROUTINE BESTX

```
      SUBROUTINE BESTX(P,Q,R,TR,H,KN,KP,KER,G,XP,Z,X)
C     THIS SUBROUTINE COMPUTES THE OPTIMUM ESTIMATE OF THE STATE VECTOR
C     FOR A NON-STATIONARY PROCESS AT EACH SAMPLE INSTANT.
     0DIMENSION P(12,12),Q(12,12),H(12,12),TR(12,12),TS(12,12),DEL(12,12
     1);G(12,12),R(12,12),GT(12,12),HT(12,12),TEMP(12,12),TEMP2(12,12)
     2,TEMP1(12,12),KER(1),XP(12,12),ZP(12,12),X(12,12)
      CALL TRANS (H,KP,KN,HT)
  107 CALL PROD (TR,G,KN,KN,KP,DEL)
      CALL PROD (DEL,H,KN,KP,KN,TEMP)
      DO 108 I=1,KN
      DO 108 J=1,KN
  108 TEMP(I,J)= -TEMP(I,J)
      CALL ADD (TR,TEMP,KN,KN,TS)
      CALL TRANS(TS,KN,KN,TEMP)
      CALL PROD (P,TEMP,KN,KN,KN,TEMP2)
      CALL PROD (TS,TEMP2,KN,KN,KN,P)
      CALL ADD(P,Q,KN,KN,P)
  104 CALL PROD (P,HT,KN,KN,KP,TEMP)
      CALL PROD (H,TEMP,KP,KN,KP,TEMP2)
      CALL ADD (TEMP2,R,KP,KP,TEMP2)
      CALL RECIP (KP,.00001,TEMP2,TEMP1,KER)
      IF (KER-2) 101,110,101
  101 CALL PROD(TEMP,TEMP1,KN,KP,KP,G)
      CALL PROD(TR,X,KN,KN,1,XP)
  110 CALL PROD(H,XP,KP,KN,1,ZP)
      DO 154 I=1,KN
  154 ZP(I,1) = -ZP(I,1)
```

```
      CALL ADD (Z,ZP,KP,1,TEMP)
      CALL PROD(G,TEMP,KN,KP,1,TEMP1)
  110 CALL ADD(XP,TEMP1,KN,1,X)
      END                      I K J
      SUBROUTINE PROD (A,B,N,M,L,C)
      DIMENSION A(12,12),B(12,12),C(12,12)
      DO 151 I=1,N
      DO 151 J=1,L
      C(I,J) =0.
      DO 151 K = 1,M
  151 C(I,J) = C(I,J) + A(I,K)*B(K,J)
      END                     I J
      SUBROUTINE ADD (A,B,N,M,C)
      DIMENSION A(12,12),B(12,12),C(12,12)
      DO 152 I=1,N
      DO 152 J=1,M
  152 C(I,J) = A(I,J) + B(I,J)
      END
      SUBROUTINE TRANS(A,N,M,C)
      DIMENSION A(12,12),C(12,12)
      DO 153 I = 1,N
      DO 153 J=1,M
  153 C(J,I) = A(I,J)
      END
      SUBROUTINE RECIP(N,EP,A,X,KER)
      DIMENSION A(12,12),X(12,12)
      DO 1 I=1,N                                                        00030
      DO 1 J=1,N                                                        00040
    1 X(I,J)=0.0                                                        00050
      DO 2 K=1,N                                                        00060
    2 X(K,K)=1.0                                                        00070
   10 DO 34 L=1,N                                                       00080
      KP=0                                                              00090
      Z=0.0                                                             00100
      DO 12 K=L,N                                                       00110
```

```
      IF(Z-ABSF(A(K,L)))11,12,12
   11 Z=ABSF(A(K,L))
      KP = K
   12 CONTINUE
      IF(L-KP)13,20,20
   13 DO 14 J=L,N
      Z=A(L,J)
      A(L,J)=A(KP,J)
   14 A(KP,J)=Z
      DO 15 J=1,N
      Z=X(L,J)
      X(L,J)=X(KP,J)
   15 X(KP,J)=Z
   20 IF(ABSF(A(L,L))-EP)50,50,30
   30 IF(L-N)31,34,34
   31 LP1=L+1
      DO 36 K=LP1,N
      IF(A(K,L))32,36,32
   32 RATIO=A(K,L)/A(L,L)
      DO 33 J=LP1,N
   33 A(K,J)=A(K,J)-RATIO*A(L,J)
      DO 35 J=1,N
   35 X(K,J)=X(K,J)-RATIO*X(L,J)
   36 CONTINUE
   34 CONTINUE
   40 DO 43 I=1,N
      II=N+1-I
      DO 43 J=1,N
      S=0.0
      IF(II-N)41,43,43
   41 IIP1=II+1
      DO 42 K=IIP1,N
   42 S=S+A(II,K)*X(K,J)
   43 X(II,J)=(X(II,J)-S)/A(II,II)
      KER=1
```

APPENDIX III   III-   SUBROUTINE BESTX(CONTINUED)

```
      RETURN
   50 KER=2
      END
      END
```

APPENDIX III   III-   SUBROUTINE BESTX(CONTINUED)

B. SUBROUTINE GANDPR

PURPOSE: This subroutine will provide, for each sample instant, an optimum gain matrix, and, the predicted values of the process and the observation state vectors, if the process is linear and the random excitation and corruptive measurement noise vectors have gaussian distributions.

USAGE:

1. Calling Sequence

```
CALL GANDPR (P, Q, R, TR, H, X, XP, ZP,
KN, KP, KER, G)
```

2. Arguments

   a. P - initial covariance matrix of system states - dimension (12 x 12)

   b. Q - covariance matrix of states due to gaussian excitation - dimension (12 x 12)

   c. R - covariance matrix of measurement noise - dimension (12 x 12)

   d. TR - transition matrix of process - dimension (12 x 12)

   e. H - matrix which defines the observable states - dimension (12 x 12)

f. X - optimal estimate of process state vector - dimension (12 x 12)

g. XP - predicted estimate of process state vector - dimension (12 x 12)

h. ZP - predicted estimate of observable state vector - dimension (12 x 12)

i. KN - number of states in the system - dimension (scaler)

j. KP - number of observable states - dimension (scaler)

k. KER - error indication (= 2 implies the inverse of a matrix could not be obtained) - dimension (1)

l. G - optimal gain matrix - dimension (12 x 12)

3. Equipment Configuration: CDC 1604 with FORTRAN 60.

4. Cautions to User:

a. All arguments in the main program must be dimensioned the same as those in GANDPR.

b. The main program should contain an ERROR TEST (see 2h.)

c. See attached flow chart depicting typical usage.

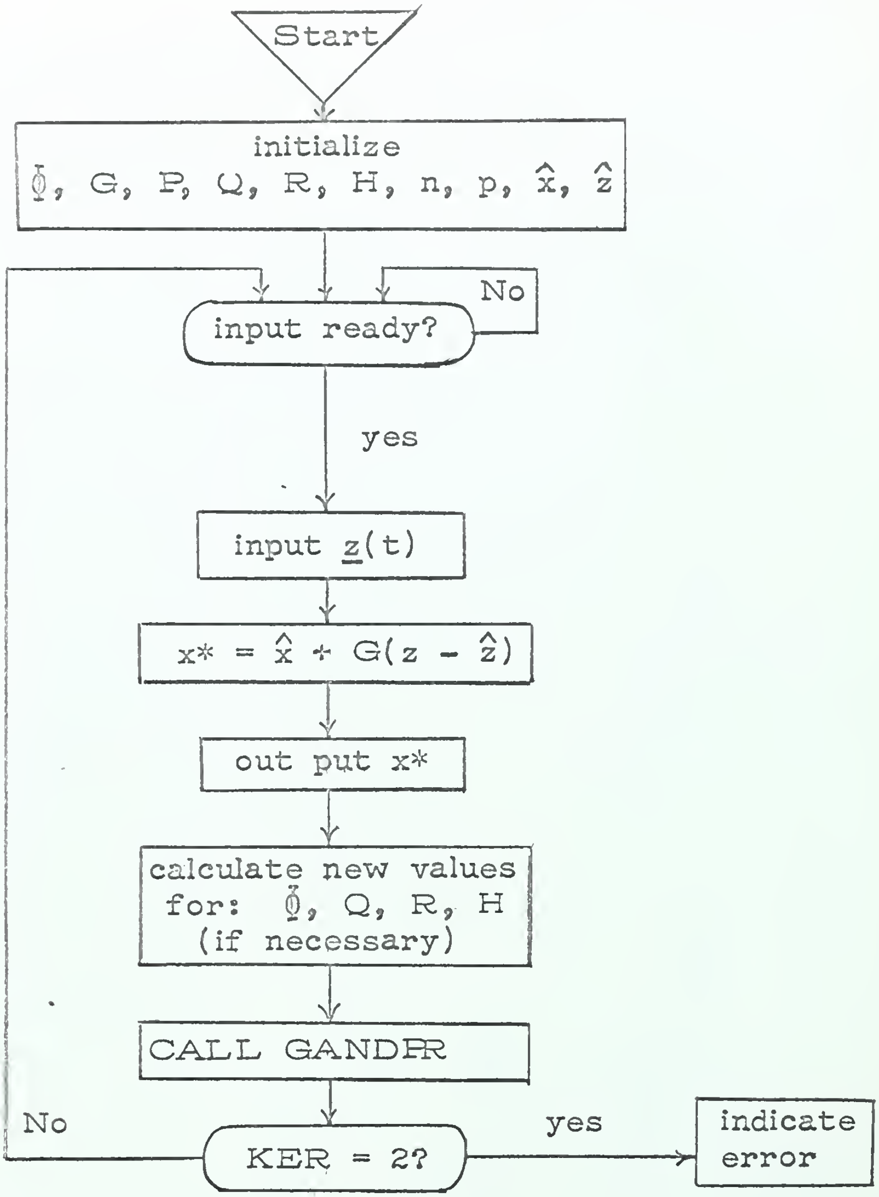

Fig. 111-2 Flow diagram showing typical usage of GANDPR

SUBROUTINE GANDPR

```
      SUBROUTINE GANDPR(P,Q,R,TR,H,X,XP,ZP,KN,KP,KER,G)
C     THIS SUBROUTINE COMPUTES THE OPTIMUM GAIN MATRIX AND THE PREDICTED
C     ESTIMATES FOR BOTH THE STATE VECTOR AND THE OBSERVABLE STATE VECTOR
C     FOR THE NON-STATIONARY PROCESS FOR EACH SAMPLE PERIOD.
     0DIMENSION P(12,12),Q(12,12),H(12,12),TR(12,12),TS(12,12),DEL(12,12
     1),G(12,12),R(12,12),GT(12,12),HT(12,12),TEMP(12,12),TEMP2(12,12)
     2,TEMP1(12,12),KER(1),XP(12,12),ZP(12,12),X(12,12)
      CALL TRANS (H,KP,KN,HT)
  107 CALL PROD (TR,G,KN,KN,KP,DEL)
      CALL PROD (DEL,H,KN,KP,KN,TEMP)
      DO 108 I=1,KN
      DO 108 J=1,KN
  108 TEMP(I,J)= -TEMP(I,J)
      CALL ADD (TR,TEMP,KN,KN,TS)
      CALL TRANS(TS,KN,KN,TEMP)
      CALL PROD (P,TEMP,KN,KN,KN,TEMP2)
      CALL PROD (TS,TEMP2,KN,KN,KN,P)
      CALL ADD(P,Q,KN,KN,P)
  104 CALL PROD (P,HT,KN,KN,KP,TEMP)
      CALL PROD (H,TEMP,KP,KN,KP,TEMP2)
      CALL ADD (TEMP2,R,KP,KP,TEMP2)
      CALL RECIP (KP,.00001,TEMP2,TEMP1,KER)
      IF (KER-2) 101,110,101
  101 CALL PROD(TEMP,TEMP1,KN,KP,KP,G)
      CALL PROD(TR,X,KN,KN,1,XP)
  110 CALL PROD(H,XP,KP,KN,1,ZP)
      END
```

H^T

```
      SUBROUTINE PROD (A,B,N,M,L,C)
      DIMENSION A(12,12),B(12,12),C(12,12)
      DO 151 I=1,N
      DO 151 J=1,L
      C(I,J) =0.
      DO 151 K = 1,M
  151 C(I,J) = C(I,J) + A(I,K)*B(K,J)
      END
      SUBROUTINE ADD (A,B,N,M,C)
      DIMENSION A(12,12),B(12,12),C(12,12)
      DO 152 I=1,N
      DO 152 J=1,M
  152 C(I,J) = A(I,J) + B(I,J)
      END
      SUBROUTINE TRANS(A,N,M,C)
      DIMENSION A(12,12),C(12,12)
      DO 153 I = 1,N
      DO 153 J=1,M
  153 C(J,I) = A(I,J)
      END
      SUBROUTINE RECIP(N,EP,A,X,KER)
      DIMENSION A(12,12),X(12,12)
      DO 1 I=1,N
      DO 1 J=1,N
    1 X(I,J)=0.0
      DO 2 K=1,N
    2 X(K,K)=1.0
   10 DO 34 L=1,N
      KP=0
      Z=0.0
      DO 12 K=L,N
      IF(Z-ABSF(A(K,L)))11,12,12
   11 Z=ABSF(A(K,L))
      KP = K
   12 CONTINUE
```

APPENDIX III   III-   SUBROUTINE GANDPR(CONTINUED)

```
      IF(L-KP)13,20,20
   13 DO 14 J=L,N
      Z=A(L,J)
      A(L,J)=A(KP,J)
   14 A(KP,J)=Z
      DO 15 J=1,N
      Z=X(L,J)
      X(L,J)=X(KP,J)
   15 X(KP,J)=Z
   20 IF(ABSF(A(L,L))-EP)50,50,30
   30 IF(L-N)31,34,34
   31 LP1=L+1
      DO 36 K=LP1,N
      IF(A(K,L))32,36,32
   32 RATIO=A(K,L)/A(L,L)
      DO 33 J=LP1,N
   33 A(K,J)=A(K,J)-RATIO*A(L,J)
      DO 35 J=1,N
   35 X(K,J)=X(K,J)-RATIO*X(L,J)
   36 CONTINUE
   34 CONTINUE
   40 DO 43 I=1,N
      II=N+1-I
      DO 43 J=1,N
      S=0.0
      IF(II-N)41,43,43
   41 IIP1=II+1
      DO 42 K=IIP1,N
   42 S=S+A(II,K)*X(K,J)
   43 X(II,J)=(X(II,J)-S)/A(II,II)
      KER=1
      RETURN
   50 KER=2
      END
      END
```

APPENDIX III   III-   SUBROUTINE GANDPR(CONTINUED)

thesJ294
Optimal filter design for sampled data s
3 2768 002 10029 9
DUDLEY KNOX LIBRARY